

# SIKIŞTIRMA ALGORİTMALARI PROJESİ

Emirhan EKŞİ, Erdem ÖZOĞLU

180202079, 180202094

emiirhaneksi@gmail.com , erdemozoglu@gmail.com

## Kocaeli Üniversitesi Bilgisayar Mühendisliği Bölümü Programlama Lab.2 – Proje 2

### 1. Giriş

Bu projede hedefimiz verilen dosyanın içeriğini istenilen sıkıştırma algoritmalarını kullanarak sıkıştırmak ve sıkıştırılan verileri kıyaslayarak hangi algoritmanın daha başarılı olduğu sonucuna ulaşmaktır. Projemizi geliştirirken LZ77 ve Deflate Sıkıştırma Algoritmaları kullanılmıştır.

### 2. Proje Tasarımı

#### 2.1 Proje geliştirme ortamı

Projemiz C programlama dili ile CodeBlocks IDE üzerinde geliştirilmiştir.

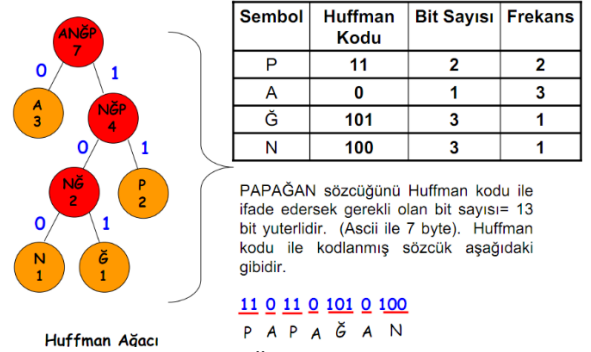
#### 2.2 Kullandığımız Algoritmalar

##### 2.2.1 LZ77 Sıkıştırma Algoritması

LZ77 Algoritması 1977 yılında Abraham Lembel ve Jacob Ziv tarafından bulunmuştur. Algoritma ismini burdan almaktadır. Algoritmanın temel olarak hedeflediği metin dosyalarında tekrar eden kısımları ortadan kaldırmaktadır. LZ77 algoritmasında token ismini verdiğimiz üç parametre alan yapılar kullanılır. Bu yapıların kullanılmasında hedeflenen tekrar eden kelime gruplarının tekrarını önlemektir. Kullanılan tokenlar (0,0,"veri") şeklinde olmalıdır. Burdaki veri okunan karakterden bir sonraki karakteri temsil etmektedir.

##### 2.2.2 DEFLATE Sıkıştırma Algoritması

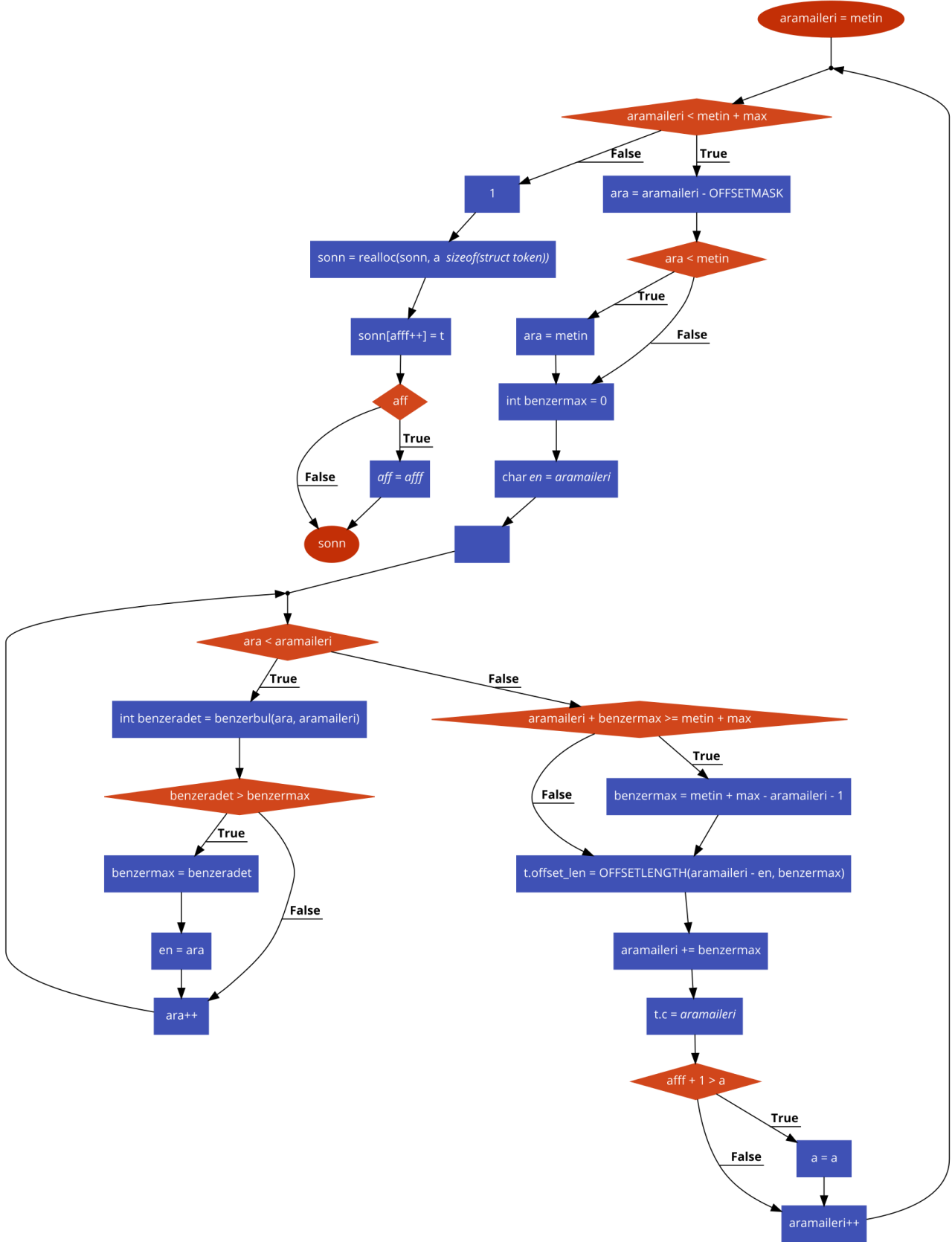
Bir diğer algoritmamız DEFLATE sıkıştırma algoritması. Bu algoritmayı Huffman ve LZSS algoritmalarının bileşimi gibi düşünebiliriz.



Resim 1 : Huffman Ağacı Örneği

Biz projemizde Deflate Algoritmasını geliştirirken sırasıyla önce Huffman algoritmasını daha sonra LZSS algoritmasını uyguladık. LZSS algoritması ile LZ77 algoritmasının farkı kısaca şu şekildedir: LZ77 algoritmasında tokenler (0,0,"veri") şeklinde 3 parametre almaktadır. LZSS algoritmasında ise okunan karakterden sonraki karakter tokenda tutulmadığından LZSS algoritması iki parametre almaktadır.

## 2.3 Akış Diyagramı



## 2.4 Projenin Görüntüleri

```
DOSYANIZ BASARILI BIR SEKILDE SIKISTIRILIP output.txt OLARAK DOSYA KONUMUNA YAZDIRILMISTIR.  
Orjinal Boyut: 4.20 KB  
Encode Boyutu: 1.70 KB  
Basari Orani: %40.47  
  
Process returned 0 (0x0)   execution time : 0.067 s  
Press any key to continue.
```

*Resim 2 : LZ77 algoritması başarılı çalıştığında*

```
Projenin konumunda metin.txt adli bir dosyaniz bulunmamaktadır.  
Process returned -1073741819 (0xC0000005)   execution time : 1.818 s  
Press any key to continue.
```

*Resim 3 : Dosya yoksa uyarı mesajı*

## 2.5 Proje Geliştirirken Karşılaşılan Zorluklar

- Her ne kadar Huffman Algoritmasını Veri Yapıları dersinde teorik olarak öğrensek de proje geliştirirken pratik olarak nasıl uygulamaya döneceğimizi bilmiyorduk.
- LZ77 Algoritması ve Deflate Algoritması hakkında kaynak bulmakta oldukça zorlandık, aynı zamanda bu algoritmalar ilk defa karşılaştığımız algoritmalar.
- Deflate Algoritmasını geliştirirken belli kaynaklarda LZ77+Huffman belli kaynaklarda LZSS+Huffman olduğunu gördük. Bu karmaşıklık da bizim için zorlayıcı oldu. Son olarak Deflate algoritması için Huffman+LZSS'e karar verdik. Ayrıca LZSS ile LZ77 arasındaki farkın token'larda bir parametre eksik olduğunu öğrendik.

## 2.6 Algoritma Analizi ve Sözde Kod

LZ77 Algoritması için :

1. Sıkıştırılması istenen txt dosyasını fopen komutuyla oku.
2. Metinden bir harf al
3. Sözlükte harfi ara
4. Metindeki sıradaki harfi aldığı anda sözlükteki bir kayda karşılık geliyorsa metinden harf almaya devam et
5. Şimdiye kadar uyan sözlük değerini sonuca bas
6. Uyumu bozan yeni harfle birlikte şimdiye kadar uyan sözlük kaydını, yeni sözlük kaydı olarak ekle
7. Metin bitmediyse 2. Adımdan uymayan bu yeni harf ile devam et.
8. Metin bittiğinde sıkıştırılmış veriyi dosyaya yazdır.

## 2.7 Proje Kazanımları

- Dosya işlemleri kullanarak C programlama dilinde proje geliştirmek.
- Deflate sıkıştırma algoritmasının çalışma yapısını öğrenmek ve Deflate ile uygulama geliştirmek.
- LZ77 sıkıştırma algoritmasının çalışma yapısını öğrenmek ve LZ77 ile uygulama geliştirmek.
- Sıkıştırma algoritmalarının hangisinin daha verimli olduğunu proje geliştirme aracılığı ile doğrudan öğrenmek.
- LZ77 ve LZSS algoritmalarının arasındaki farkı öğrenmek.

## 3. Kaynakça

- <https://ysar.net/algoritma/lz77.html>, erişim : 17.05.2020
- Ramazan Akbuz  
<https://ramazanakbuz.com/lz77-lz88-sikistirma-algoritmasi/>  
erişim : 17.05.2020
- <https://ysar.net/algoritma/huffman-kodlamasi.html>  
erişim : 17.05.2020
- Şadi Evren Şeker  
<http://bilgisayarkavramlari.sadievrenseker.com/2009/02/25/huffman-kodlamasi-huffman-encoding/>  
erişim . 17.05.2020
- Şadi Evren Şeker  
<http://bilgisayarkavramlari.sadievrenseker.com/2008/10/22/c-ile-dosya-islemleri/>  
erişim : 17.05.2020